

Submitted to *Operations Research*
manuscript (Please, provide the manuscript number!)

Bounding Procedures for Stochastic Dynamic Programs with Application to the Perimeter Patrol Problem

Myoungkuk Park

Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843, robotian@gmail.com

Krishnamoorthy Kalyanam

Infoscitex Corporation, Dayton, OH 45431, krishna.kalyanam@gmail.com

Swaroop Darbha

Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843, dswaroop@tamu.edu

Phil Chandler

Control Design & Analysis Branch, Air Force Research Laboratory, WPAFB, OH 45433, phillip.chandler@wpafb.af.mil

Meir Pachter

Electrical Engineering Department, Air Force Institute of Technology, WPAFB, OH 45433, meir.pachter@afit.edu

One often encounters the curse of dimensionality in the application of dynamic programming to determine optimal policies for controlled Markov chains. In this paper, we provide a method to construct sub-optimal policies along with a bound for the deviation of such a policy from the optimum via a linear programming approach. The state-space is partitioned and the optimal cost-to-go or value function is approximated by a constant over each partition. By minimizing a non-negative cost function defined on the partitions, one can construct an approximate value function which also happens to be an upper bound for the optimal value function of the original Markov Decision Process (MDP). As a key result, we show that this approximate value function is *independent* of the non-negative cost function (or state dependent weights as it is referred to in the literature) and moreover, this is the least upper bound that one can obtain once the partitions are specified. Furthermore, we show that the restricted system of linear inequalities also embeds a family of MDPs of lower dimension, one of which can be used to construct a lower bound on the optimal value function. The construction of the lower bound requires the solution to a combinatorial problem. We apply the linear programming approach to a perimeter surveillance stochastic optimal control problem and obtain numerical results that corroborate the efficacy of the proposed methodology.

Key words: Stochastic Dynamic Programs, Linear Programming, State Aggregation

1. Introduction

The Linear Programming (LP) approach to solving dynamic programs (DPs) originated from the papers: Manne (1960), d'Epenoux (1963), Denardo (1970), Hordijk and Kallenberg (1979). The basic feature of an LP approach for solving DPs corresponding to maximization of a discounted payoff is that the optimal solution of the DP (also referred to as the optimal value function) is the optimal solution of the LP for *every* non-negative cost function. The constraint set describing the feasible solution of the LP and the number of independent variables are typically very large (*curse of dimensionality*) and hence, obtaining the exact solution of a DP (stochastic or otherwise) via an LP approach is not practical. Despite this limitation, an LP approach provides a tractable method for approximate dynamic programming (Mendelsohn 1980, Schweitzer and Seidmann 1985, Trick and Zin 1997) and the advantages of this approach may be summarized as follows:

1. One can restrict the value function to be of a certain parameterized form, thereby reducing the dimension of the LP to the size of the parameter set to make it tractable.
2. The solution to the LP provides upper bounds for the value function (lower bounds, if minimizing a discounted cost, as opposed to maximizing discounted payoff, is considered as the optimization criteria).

The main questions regarding the tractability and quality of approximate DP revolve around restricting the value function in a suitable way. The questions are: (1) How does one restrict the value function, i.e., what basis functions should one choose for parameterizing the value function? (2) Are there any (a posteriori) bounds that one can provide about the value function from the solution of a restricted LP? If the restrictions imposed on the value function are consistent with the physics/structure of the problem, one can expect reasonably tight bounds. There is another question that naturally arises: In the unrestricted case, the optimal solution of the LP is independent of the choice of the non-negative cost function. While it is unreasonable to expect that the optimal value function be a feasible solution of the restricted LP, one can ask if the optimal solution of the restricted LP is the same for *every* choice of non-negative cost function for the LP. It has been reported in the literature that this is unfortunately not the case (De Farias and Van Roy 2003).

If the LP is not properly restricted, it can lead to poor approximation and perhaps, even infeasibility (Gordon 1999). A common approach is to approximate the value (cost-to-go) function by a linear functional of a priori chosen basis functions (Schweitzer and Seidmann 1985). This approach is attractive in that for a certain class of basis functions, feasibility of the approximate (or restricted) LP is guaranteed (De Farias and Van Roy 2003). A straightforward method for selecting the basis functions is through a state aggregation method. Here the state space is partitioned into disjoint sets or partitions and the approximate value function is restricted to be the same for all the states in a partition. The number of variables for the LP therefore reduces to the number of partitions. State aggregation based approximation techniques were originally proposed by Axsäter (1983), Bean et al. (1987), Mendelssohn (1982). Since then, substantial work has been reported in the literature on this topic (see Van Roy (2006) and the reference therein). In this article, we adopt the state aggregation method.

Although imposing restrictions on the value function reduces the size of the restricted LP, the number of constraints does not change. Since the number of constraints is at least of the same order as the number of states of the DP, one is faced with a restricted LP with a large number of constraints. An LP with a large number of constraints may be solved if there is an automatic way to separate a non-optimal solution from an optimal one (Grötschel et al. 1981); otherwise, one may have to resort to heuristics or settle for an approximate solution. Separation of a non-optimal solution from an optimal one is easier if one has a compact representation of constraints (Morrison and Kumar 1999) or if a subset of the constraints that dominate other constraints can easily be identified from the structure of the problem (Krishnamoorthy et al. 2011b). Heuristic methods include aggregation of constraints, sub-sampling of constraints (De Farias and Van Roy 2003), constraint generation methods (Grötschel and Holland 1991, Schuurmans and Patrascu 2001) and other approaches (Trick and Zin 1993).

If the solution of the restricted LP is the same for *every* non-negative cost function of the LP, then it suggests that the constraint set for the restricted LP embeds the constraint set for the exact LP corresponding to a reduced order Markov Decision Process (MDP). If one adopts a naive

approach and “aggregates” every state into a separate partition, we obtain the original exact LP and clearly, for this LP, the solution is independent of the non-negative cost function. It would seem reasonable to expect that this would generalize to partitions of arbitrary size and in fact, we prove this to be the case in this article. One can construct a sub-optimal policy from the solution to the restricted LP by considering the policy that is greedy with respect to the approximate value function (Porteus 1975). By construction, the expected discounted payoff for the sub-optimal policy will be a lower bound to the optimal value function and hence, can be used to quantify the quality of the sub-optimal policy. Also the lower bound will be closer to the optimal value function than the approximate value function by virtue of the monotonicity property of the Bellman operator. But the lower bound computation is not efficient since the procedure involved is tantamount to policy evaluation which involves the solution to a system of linear equations of the same size as the state-space. In this work, we have developed a novel disjunctive LP, whose solution can be used to construct a lower bound to the optimal value function. The contributions of our work may be summarized as follows:

- If one were to adopt a state aggregation approach, then the solution to the restricted LP is shown to be *independent* of the non-negative cost function. Moreover, the optimal solution is dominated by every feasible solution to the restricted LP.
- We also show that considering alternate LP formulations via lifting of variables or by considering a bigger feasible set via iterated Bellman inequalities (Wang and Boyd 2010) does not improve upon the upper bound provided by the restricted LP.
- A subset of the constraints of the restricted LP can be used for constructing a lower bound for the optimal value function. However, this involves solving a disjunctive LP, which may not be computationally tractable.
- We demonstrate the use of aggregation based restricted LPs for a perimeter surveillance stochastic control problem. For the application considered here, we show that both the lower bounding disjunctive LP and the upper bounding restricted LP can be solved efficiently since they both reduce to exact LPs corresponding to some lower dimensional MDPs.

The rest of the paper is organized as follows: we provide a general overview of stochastic dynamic programs in section 2 followed by LP preliminaries in section 2.1. In section 3, we introduce the aggregation method and discuss the restricted LP approach that can be used to approximate the optimal value function. In the same section, we also present a novel disjunctive LP that can be used to compute a lower bound to the optimal value function. We introduce the perimeter alert patrol problem in section 4 and also elaborate on the efficient LP formulations that arise out of the structure in the problem. We corroborate the structure in the perimeter patrol problem via numerical results in section 5. Finally, we support the proposed approximation methodology via simulation results in section 5.1, followed by summary in section 6. Supplementary material and lengthy proofs, that have been left out of the main body of the paper, for clarity, have been included in the Appendix.

2. Stochastic Dynamic Programming

Consider a discrete-time Markov decision process (MDP) with a finite state space $\mathcal{S} = \{1, 2, \dots, |\mathcal{S}|\}$. For each state $x \in \mathcal{S}$, there is a finite set of available actions \mathcal{U}_x . From current state x , taking action $u \in \mathcal{U}_x$ under the random influence Y results in a reward $R_u(x)$. The system follows some discrete-time dynamics given by:

$$x(t+1) = f(x(t), u(t), Y(t)), \quad (1)$$

where t indicates time. We assume that the random input Y can only take a finite set of values $Y_l; l = 0, \dots, m$ and there is a probability associated with each choice p_l . State transition probabilities $P_u(x, y)$ represent, for each pair (x, y) of states and each action $u \in \mathcal{U}_x$, the probability that the next state will be y given that the current state is x and the current action taken is u i.e.,

$$P_u(x, y) = \begin{cases} 0, & \text{if } y \neq f(x, u, Y_l) \text{ for any } l \in \{0, \dots, m\}, \\ \sum_{j \in \mathcal{C}} p_j, & \text{where } \mathcal{C} = \{l | y = f(x, u, Y_l)\}. \end{cases} \quad (2)$$

Any *stationary* policy, π , specifies for each state $x \in \mathcal{S}$, a control action $u = \pi(x)$. We abuse notation and also write the transition probability matrix associated with policy π to be P_π , where $P_\pi(x, y) = P_{\pi(x)}(x, y)$. Similarly, we express the column vector of immediate payoffs associated with the policy

π to be R_π , where $R_\pi(x) = R_{\pi(x)}(x)$. We are interested in solving a stochastic control problem, which amounts to selecting a policy that maximizes the infinite-horizon discounted reward of the form,

$$V_\pi(x_0) = \mathbf{E} \left[\sum_{t=0}^{\infty} \lambda^t R_\pi(x(t)) \middle| x(0) = x_0 \right],$$

where $\lambda \in [0, 1)$ is a temporal discount factor. We obtain the optimal policy by solving Bellman's equation,

$$V^*(x) = \max_{u \in \mathcal{U}_x} \left\{ R_u(x) + \lambda \sum_{l=0}^m p_l V^*(f(x, u, Y_l)) \right\}, \forall x \in \mathcal{S}, \quad (3)$$

where, $V^*(x)$ is the optimal value function (or optimal discounted payoff) starting from state x .

The optimal policy then is given by,

$$\pi^*(x) = \arg \max_{u \in \mathcal{U}_x} \left\{ R_u(x) + \lambda \sum_{l=0}^m p_l V^*(f(x, u, Y_l)) \right\}, \forall x \in \mathcal{S}. \quad (4)$$

The Bellman equation (3) can be solved using standard DP methods such as value iteration (Howard 1960) or policy iteration (Bellman 1957); however, it is computationally not tractable, if the size of state space considered is unmanageably large. For this reason, one is interested in tractable approximate methods that yield suboptimal solutions with some guarantees on the deviation of the associated approximate value function from the optimal one.

2.1. Linear Programming Approach

In this subsection, we briefly touch upon two lemmas that we will use in the subsequent sections. Bellman's equation suggests that the optimal value function satisfies the following set of linear inequalities, which we will refer to as the Bellman inequalities:

$$\begin{aligned} V(x) &\geq R_u(x) + \lambda \sum_{l=0}^m p_l V(f(x, u, Y_l)), \forall u \in \mathcal{U}_x, \forall x \in \mathcal{S}. \\ \Leftrightarrow V &\geq R_u + \lambda P_u V, \forall u. \end{aligned} \quad (5)$$

Consider any integer $L \geq 1$ and for $j = 1, 2, \dots, L$, let V_j be a vector satisfying a generalization of the Bellman inequalities, referred to as the iterated Bellman inequalities (Wang and Boyd 2010):

$$V_{j+1}(x) \geq R_u(x) + \lambda \sum_{l=0}^m p_l V_j(f(x, u, Y_l)), \forall x, u, \quad \forall j = 1, 2, \dots, L-1, \quad (6)$$

$$V_1(x) \geq R_u(x) + \lambda \sum_{l=0}^m p_l V_L(f(x, u, Y_l)), \quad \forall x, u. \quad (7)$$

Clearly, when $L = 1$, the above system of inequalities collapses to the Bellman inequalities. The iterated Bellman inequalities may be compactly represented as:

$$\begin{aligned} V_{j+1} &\geq R_u + \lambda P_u V_j, \quad \forall u, \quad j = 1, 2, \dots, L-1, \\ V_1 &\geq R_u + \lambda P_u V_L, \quad \forall u. \end{aligned} \quad (8)$$

We note that the above set of inequalities have cyclic symmetry, i.e., one gets the same set of inequalities by replacing the vectors V_1, V_2, \dots, V_L by $V_2, V_3, \dots, V_L, V_1$ respectively. Let π be any stationary policy. Then we have,

$$V_{j+1} \geq R_\pi + \lambda P_\pi V_j, \quad j = 1, 2, \dots, L-1, \quad (9)$$

$$V_1 \geq R_\pi + \lambda P_\pi V_L. \quad (10)$$

By recursively applying (9) to V_L, V_{L-1}, \dots etc., in (10), we get,

$$[I - \lambda^L P_\pi^L] V_1 \geq [I + \lambda P_\pi + \dots + \lambda^{L-1} P_\pi^{L-1}] R_\pi, \quad \forall \pi.$$

By cyclic symmetry, every $V_j, j = 2, 3, \dots, L$, also satisfies the above inequality.

LEMMA 1. *Let the vector V satisfy the following set of inequalities:*

$$[I - \lambda^L P_\pi^L] V \geq [I + \lambda P_\pi + \dots + \lambda^{L-1} P_\pi^{L-1}] R_\pi, \quad \forall \pi. \quad (11)$$

Then, we have $V \geq V^$.*

REMARK 1. We readily see that every feasible solution of the system of inequalities (5) or (8) is lower bounded by the optimal value function V^* . By cyclic symmetry, we conclude that every feasible $V_j, j = 1, \dots, L$ is also lower bounded by V^* .

The following result relates the optimal value function to the optimal solution of an LP with a non-negative cost function and constraints of the form given by the Bellman inequalities (5) or iterated Bellman inequalities (8).

LEMMA 2. Let c be a vector of state-dependent weights with $c(x) \geq 0$ for every $x \in \mathcal{S}$. Then V^* minimizes the linear functional $c^T V$ among all V 's satisfying the Bellman inequalities (5). Correspondingly, the L -tuple (V^*, \dots, V^*) minimizes the linear functional $\sum_{j=1}^L c^T V_j$ among all L -tuples (V_1, \dots, V_L) satisfying the iterated Bellman inequalities (8).

Proof of Lemma 2. The proof follows from the fact that $V \geq V^*$ and hence, $c^T(V - V^*) \geq 0$. Since V^* is feasible for the inequalities (8) for any $L \geq 1$, the result follows. Similarly, since the L -tuple (V^*, \dots, V^*) is feasible for (8) and since $V_j \geq V^*$ for $j = 1, 2, \dots, L$, it readily follows that the L -tuple is optimal. \square

3. Bounds using Partitioning

Let the set of all states \mathcal{S} be partitioned into M disjoint sets, $\mathcal{S}_i, i = 1, \dots, M$. We will call the set \mathcal{S}_i the i^{th} partition. Henceforth, we will use the following notation: if $f(x, u, Y_s)$ represents the state the system transitions to starting from x and subject to a control input u and a stochastic disturbance Y_s , then $\bar{f}(x, u, Y_s)$ represents the partition to which the final state belongs. For a given u and partition index i , we define the tuple $z_x^{i,u} = (\bar{f}(x, u, Y_0), \bar{f}(x, u, Y_1), \dots, \bar{f}(x, u, Y_m))$ for every $x \in \mathcal{S}_i$. We denote by $\mathcal{T}(i, u)$ the set of all distinct $z_x^{i,u}$ for a given partition index i and control u .

3.1. Restricted Linear Program

We have, from Lemma 2, that the optimal solution to the following LP,

$$ELP := \min c^T V, \quad \text{subject to} \quad (12)$$

$$V \geq R_u + \lambda P_u V, \quad \forall u,$$

referred to as the “exact LP” in the literature, is the optimal value function V^* . Let us start with restricting the exact LP by requiring further that $V(x) = v(i)$ for all $x \in \mathcal{S}_i, i = 1, \dots, M$. Augmenting these constraints to the exact LP, one gets the following restricted LP.

$$RLP := \min \sum_{i=1}^M \sum_{x \in \mathcal{S}_i} c(x) v(i) \quad \text{subject to} \quad (13)$$

$$v(i) \geq R_u(x) + \lambda \sum_{l=0}^m p_l v(\bar{f}(x, u, Y_l)), \quad \forall x \in \mathcal{S}_i, i = 1, \dots, M, \forall u.$$

The restricted LP can also be written in the following compact form:

$$RLP = \min c^T \Phi v \quad \text{subject to} \quad (14)$$

$$\Phi v \geq R_u + \lambda P_u \Phi v, \quad \forall u,$$

where the columns of Φ (commonly referred to as “basis functions” in the literature) are given by,

$$\Phi(x, i) = \begin{cases} 1, & \text{if } x \in \mathcal{S}_i, \\ 0, & \text{otherwise.} \end{cases}, \quad i = 1, \dots, M. \quad (15)$$

The restricted LP typically deals with a much smaller number of variables i.e., $M \ll |\mathcal{S}|$. An approximate value function can be constructed from every feasible solution to RLP according to $V_{up} = \Phi v \Rightarrow V_{up}(x) = v(i), \forall x \in \mathcal{S}_i, i = 1, \dots, M$. Since the approximate value function satisfies, by construction, the Bellman inequalities (5), it is automatically an upper bound to V^* by Lemma 1. So, if v^* is the optimal solution to RLP (13), then clearly, $\Phi v^* \geq V^*$. Now we are ready to address one of the main results of the paper.

THEOREM 1. *The optimal solution, v^* , to the RLP is independent of the cost vector c once the partitions are specified.*

Proof of Theorem 1. The main idea behind the proof is the following: The constraints in the restricted LP (13) do not, in general, correspond to those of a Markov Decision Process (MDP) because the transition from one partition to another for a given control u and random input Y_l is not specified unambiguously. This is because different states in the same partition can transition to different partitions for the same u and Y_l . If one were to think of a “random” selector for a state in a partition, then the specification of u, Y_l together with the random selector specifies exactly which partition the system would transition to next, from the current partition. Let us specify the probability of picking a state in a partition, corresponding to the random selector, via the optimal dual variables for RLP . For a given partition index i , the RLP specifies a constraint on $v(i)$ for

each $x \in \mathcal{S}_i$ and u . Let the dual variable corresponding to this constraint be $\mu_u^i(x) \geq 0$ and the corresponding optimal dual variable be $\bar{\mu}_u^i(x)$. With this definition, we can proceed to prove the result via the following steps:

1. We show that for every partition index i , there is a u such that $\bar{\mu}_u^i(x) > 0$ for some $x \in \mathcal{S}_i$. This is necessary for constructing a MDP of reduced dimension in the next step; otherwise, the corresponding value of $v(i)$ is not lower bounded.

2. We define a reduced order MDP on the partitions with immediate reward and transition probability given by,

$$r_u(i) = \sum_{x \in \mathcal{S}_i} h_u^i(x) R_u(x) \text{ and } \tilde{P}_u(i, j) := \begin{cases} \sum_{x \in \mathcal{S}_i} h_u^i(x) \sum_{y \in \mathcal{S}_j} P_u(x, y), & \text{if } u \in \mathcal{U}_i, \\ 0, & \text{otherwise,} \end{cases}$$

where $u \in \mathcal{U}_i$ if $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. We may interpret the term $h_u^i(x) = \frac{\bar{\mu}_u^i(x)}{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x)}$ as the probability of picking the state x from the partition \mathcal{S}_i .

3. We show that the so-called ‘‘surrogate LP’’ obtained by aggregating the constraints of *RLP* via the optimal dual variables,

$$\begin{aligned} SLP(\bar{\mu}) := \min & \sum_{i=1}^M \underbrace{\sum_{x \in \mathcal{S}_i} c(x) v(i)}_{\bar{c}(i)}, \quad \text{subject to} \\ v(i) & \geq r_u(i) + \lambda \sum_{j=1}^M \tilde{P}_u(i, j) v(j), \quad \forall u \in \mathcal{U}_i, i = 1, \dots, M, \end{aligned} \tag{16}$$

is the exact LP corresponding to the reduced order MDP defined in step 2 above. In essence, for a given c , the optimal value function of the reduced order MDP is the optimal solution of *RLP*. We use the properties of surrogate duality (Greenberg and Pierskalla 1970, Glover 1975, 1968) to demonstrate that $SLP(\bar{\mu}) = RLP$.

4. Finally, to show that the optimal solution to *RLP* is independent of c , we note that the constraints of $SLP(\bar{\mu})$ are obtained by taking convex combinations of the constraints in *RLP*. Hence, any feasible solution to *RLP* is also feasible for $SLP(\bar{\mu})$. Since every feasible solution of the exact LP corresponding to an MDP dominates the optimal solution (from Lemma 1), we conclude

that the optimal solutions corresponding to two different cost functions c_1 and c_2 necessarily dominate each other and hence, have to be the same. \square

We shall now establish the surrogate LP result via the following lemma with the proof provided in the Appendix.

LEMMA 3. Consider a surrogate LP for the RLP through a set of dual variables, μ given by:

$$SLP(\mu) := \min \bar{c}^T v, \quad \text{subject to} \quad (17)$$

$$\sum_{x \in \mathcal{S}_i} \mu_u^i(x) v(i) \geq \sum_{x \in \mathcal{S}_i} \mu_u^i(x) \left[R_u(x) + \lambda \sum_{l=0}^m p_l v(\bar{f}(x, u, Y_l)) \right], \quad \forall u, i = 1, \dots, M.$$

Then, $\exists \bar{\mu} \geq 0$ such that, $SLP(\bar{\mu}) = RLP$, and, for every partition index $i = 1, \dots, M$, $\exists u$ such that $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. Moreover, the optimal solution v^* to RLP is independent of the cost vector \bar{c} and any other feasible solution v to RLP dominates v^* .

Theorem 1 implies that the upper bound for the optimal value function cannot be improved by changing the cost function from a linear to a non-linear function or by restricting the feasible set of RLP further since the optimal solution of RLP is dominated by every feasible solution of RLP. Also Φv^* is the least upper bound to the optimal value function V^* since any other feasible v to RLP satisfies $\Phi v \geq \Phi v^*$. Hence, a refinement of the upper bound must necessarily involve an enlargement of the feasible set if one wants to stick to an LP formulation, i.e., it should include the feasible set of (13) and possibly other tighter upper bounds than the optimal solution of RLP. Lifting of variables is one way to improve the bound; in this connection, we show in the following section that neither a general lifted LP nor one obtained by including the iterated Bellman inequalities in the constraint set improves the upper bound.

REMARK 2. If one considers the sub-optimal dual variables, $\mu_u^i(x) = \frac{1}{|\mathcal{S}_i|}, \forall x \in \mathcal{S}_i, \forall u$, then solving the corresponding surrogate dual, $SLP(\mu)$, to obtain an approximate value function, would result in the so-called “hard aggregation” method (see Sec. 4 of Bertsekas (2007)).

REMARK 3. When μ and Φ are allowed to have arbitrary positive entries satisfying $\sum_{x=1}^{|\mathcal{S}|} \mu_u^i(x) = 1, \forall i \in \{1, \dots, M\}$ and $\sum_{j=1}^M \Phi(y, j) = 1, \forall y \in \mathcal{S}$, the method is referred to as “soft aggregation”

(Singh et al. 1995). Unfortunately, in this case, the optimal solution to the restricted LP formulation (14) has been shown to be dependent on the cost function (De Farias and Van Roy 2003).

3.2. Lifted Restricted Linear Programs

It may appear that we can get tighter upper bounds than those provided by the *RLP* by considering either lifted LPs whose feasible set is larger than that of *RLP* or LPs with a different objective function. We will show, in this section, that unfortunately this is not the case. In general, one can construct a lifted LP of the form:

$$LLP := \min \bar{c}^T v + d^T z, \quad \text{subject to}$$

$$V(x) \geq R_u(x) + \lambda \sum_{l=0}^m p_l(V(f(x, u, Y_l))), \quad \forall x, u, \quad (18)$$

$$V(x) = v(i), \quad \forall x \in \mathcal{S}_i, i = 1, \dots, M, \quad (19)$$

$$z \geq 0,$$

where z is the additional vector of variables used in lifting so that the feasible set is not empty. Then, it follows that if (\tilde{v}, \tilde{z}) is optimal to *LLP*, then \tilde{v} will be a feasible solution to *RLP*. Consequently, $\Phi \tilde{v} \geq \Phi v^*$, where v^* is the optimal solution of the *RLP*. In other words, one gets no better bound via lifting if the constraints (18) and (19) are included. One could also use the iterated Bellman inequalities (8) for constructing a lifted LP of the form:

$$IB := \min \sum_{j=1}^L \bar{c}^T v_j, \quad \text{subject to}$$

$$v_{j+1}(i) \geq R_u(x) + \lambda \sum_{l=0}^m p_l v_j(\bar{f}(x, u, Y_l)), \quad \forall x \in \mathcal{S}_i, \quad \forall i, u, \quad j = 1, \dots, L-1, \quad (20)$$

$$v_1(i) \geq R_u(x) + \lambda \sum_{l=0}^m p_l v_L(\bar{f}(x, u, Y_l)), \quad \forall x \in \mathcal{S}_i, \quad \forall i, u. \quad (21)$$

Again, it turns out that the above lifted *LP* is incapable of providing a better bound, as can be seen from the following result.

THEOREM 2. *If $v_{IB} = (v_1, \dots, v_L)$ is a feasible solution to *IB*, then $v_j \geq v^*$ for $j = 1, \dots, L$, where v^* is the optimal solution to *RLP*.*

The proof for Theorem 2 follows along the lines of Lemma 3. We will construct a surrogate LP for the lifted LP (20) with the optimal dual variables of *RLP*. We immediately recognize that the inequalities defining the surrogate LP are, in fact, the iterated Bellman inequalities associated with the reduced order MDP defined in step 2 of the proof of Theorem 1. So, the result follows from Lemma 2 and Remark 1.

Proof of Theorem 2. Let $\bar{\mu}$ be the optimal dual variables to *RLP* (13). From Lemma 3, for every partition index $i \in \{1, \dots, M\}$, there exists a u such that $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. For a fixed i and u , we multiply the inequalities (20, 21) associated with a particular $x \in \mathcal{S}_i$ with $\bar{\mu}_u^i(x)$ and sum over all the $x \in \mathcal{S}_i$. Then, we get the following surrogate LP:

$$SIB := \min \sum_{j=1}^L \bar{c}^T v_j, \quad \text{subject to}$$

$$v_{j+1}(i) \geq r_u(i) + \lambda \sum_{x \in \mathcal{S}_i} h_u^i(x) \sum_{l=0}^m p_l v_j(\bar{f}(x, u, Y_l)), \quad \forall u \in \mathcal{U}_i, \forall i, \quad j = 1, \dots, L-1, \quad (22)$$

$$v_1(i) \geq r_u(i) + \lambda \sum_{x \in \mathcal{S}_i} h_u^i(x) \sum_{l=0}^m p_l v_L(\bar{f}(x, u, Y_l)), \quad \forall u \in \mathcal{U}_i, \forall i, \quad (23)$$

where, $u \in \mathcal{U}_i$ if $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. As before, the one-step reward function,

$$r_u(i) = \frac{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) R_u(x)}{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x)}, \quad \text{where, } h_u^i(x) = \frac{\bar{\mu}_u^i(x)}{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x)}, \quad \forall u \in \mathcal{U}_i.$$

By Lemma 2, the optimal solution to *SIB* is of the form $v_{SIB}^* = (v^*, \dots, v^*)$, where v^* is the optimal solution to *SLP*($\bar{\mu}$) (and by Lemma 3, also the optimal solution to *RLP*). Since any feasible solution to *IB*, $v_{IB} = \{v_1, \dots, v_L\}$ is also feasible to *SIB*, it follows, from Lemma 1, that $v_j \geq v^*$ for every $j = 1, \dots, L$. \square

So, we conclude that lifting through the use of iterated Bellman inequalities does not help in finding a tighter upper bound than the *RLP* optimal solution. Also using any other non-linear objective function will not improve the upper bound as long as the iterated Bellman inequalities (20) and (21) are included in the constraints set. In the next section, we focus our attention on the construction of a lower bound for the optimal value function.

3.3. Lower Bound for the Optimal Value Function

For any candidate approximate value function \tilde{V} , one can construct a sub-optimal “greedy” policy according to:

$$\tilde{\pi}(x) = \arg \max_u \left\{ R_u(x) + \lambda \sum_y P_u(x, y) \tilde{V}(y) \right\}, \quad \forall x \in \mathcal{S}.$$

Let us define the improvement in value function, $\tilde{\alpha}(x) := R_{\tilde{\pi}}(x) + \lambda \sum_y P_{\tilde{\pi}}(x, y) \tilde{V}(y) - \tilde{V}(x)$. Note that there is no improvement, i.e., $\tilde{\alpha} \equiv 0$, when $\tilde{V} = V^*$. The expected discounted payoff, $V_{\tilde{\pi}}$, corresponding to the suboptimal policy $\tilde{\pi}$, satisfies the following bound (Porteus 1975):

$$\tilde{V}(x) + \frac{1}{1-\lambda} \min_y \tilde{\alpha}(y) \leq V_{\tilde{\pi}}(x) \leq V^*(x), \quad \forall x \in \mathcal{S}.$$

In our experience, the lower bound to the optimal value function provided by $V_{\tilde{\pi}}$ is very conservative. Also computation of $V_{\tilde{\pi}}$ involves solving a linear system of equations of size $|\mathcal{S}|$, which would be expensive for a large state-space. So, we construct a novel alternate lower bound as follows. Recall that for each $x \in \mathcal{S}_i$, $V^*(x)$ satisfies the Bellman inequality (5):

$$\begin{aligned} V^*(x) &\geq R_u(x) + \lambda \sum_{l=0}^m p_l V^*(f(x, u, Y_l)), \quad \forall u, \\ &\geq R_u(x) + \lambda \sum_{l=0}^m p_l \min_{y \in \bar{f}(x, u, Y_l)} V^*(y), \quad \forall u. \end{aligned} \quad (24)$$

Let $\bar{w}(i) := \min_{x \in \mathcal{S}_i} V^*(x)$, $i = 1, \dots, M$. Then, it follows from (24) that,

$$\bar{w}(i) \geq \min_{x \in \mathcal{S}_i} \left\{ R_u(x) + \lambda \sum_{l=0}^m p_l \bar{w}(\bar{f}(x, u, Y_l)) \right\} \quad \forall u, i = 1, \dots, M. \quad (25)$$

The above set of inequalities motivates the following non-linear program:

$$\begin{aligned} NLP &:= \min \bar{c}^T w, \quad \text{subject to} \\ w(i) &\geq \min_{x \in \mathcal{S}_i} \left\{ R_u(x) + \lambda \sum_{l=0}^m p_l w(\bar{f}(x, u, Y_l)) \right\}, \quad \forall u, i = 1, \dots, M. \end{aligned} \quad (26)$$

Let w^* be the optimal solution to NLP . By construction, we see that \bar{w} is a feasible solution to the NLP and hence,

$$\bar{c}^T w^* \leq \bar{c}^T \bar{w} = \sum_{i=1}^M \bar{c}(i) \min_{x \in \mathcal{S}_i} V^*(x).$$

So, by choosing $\bar{c}(i) = 1$ and $\bar{c}(j) = 0$ for all $j \neq i$, one can obtain a lower bound to the optimal value function for all the states in the i^{th} partition. Moreover, if the problem under consideration exhibits a special structure, one can show that NLP collapses to an LP that can be efficiently solved. The perimeter patrol problem considered herein exhibits such a structure; we demonstrate this in the next section.

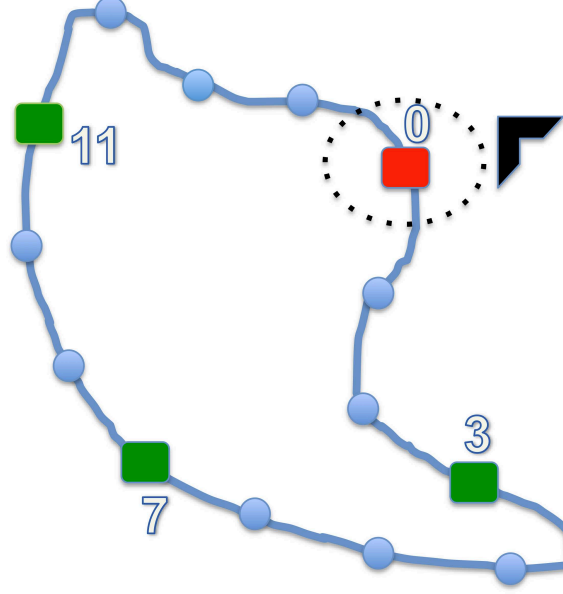
REMARK 4. The NLP is referred to as a disjunctive linear program (Balas 1979) and the optimal solution to NLP is the solution that minimizes the same linear objective function over the convex hull of the feasible solutions of NLP . Balas (1998) provides two methods to solve the problem: one through a lifted representation for the convex hull of the feasible set of NLP and the other through a cutting plane technique. Since the number of lifted variables is of $O(M^2|\mathcal{U}|)$; if $M = 10,000$, then one must deal with a lifted LP with 100 million variables. The original (non-aggregated LP) has about 10 million variables and hence, the lifted representation method is not practical. For this reason, the cutting plane technique is a viable alternate method.

REMARK 5. The lower bound provided by NLP is a non-trivial one because the optimal solution is the optimal value function of a reduced order MDP. Hence, the lower bound will be better than at least the value function associated with some suboptimal policy and so, is non-trivial and non-conservative.

REMARK 6. While \mathcal{S}_i may have a lot of states, the number of entries on the right hand side of the non-linear constraint (26) over which the minimization must be carried out is the cardinality of $\mathcal{T}(i, u)$. NLP is combinatorial in nature, in the sense that one must pick one $(m+1)$ tuple for each i and u over which the optimization must be carried out. However, for each $(m+1)$ tuple picked, one obtains an MDP. So, the system of inequalities (26) describes a family of underlying MDPs.

4. Perimeter Patrol Problem

The perimeter patrol problem arose from the Cooperative Operations in Urban Terrain (COUNTER) project at AFRL (Gross et al. 2006). In this problem, there is a perimeter which must

Figure 1 Perimeter patrol scenario with UAV loitering at alert station.

be monitored by a collection of UAVs (we will consider only one UAV here). Along the perimeter, there are m alert stations equipped with Unattended Ground Sensors (UGSs) which detect intrusions or incursions into the perimeter. For the sake of simplicity, we assume that incursions into the perimeter can only occur at the stations. An incursion could be a nuisance (false alarm) or a real threat. The UGS raise an alarm or an alert whenever there is an incursion. The camera equipped UAV responds to an alert by flying to the alert site and loitering there, while a remotely located operator steers the gimbaled camera looking for the source of the alarm. Here the operator serves the role of a classifier or a sensor, i.e., the operator must determine, from the video information, whether the intrusion is a nuisance or a threat. For details on the perimeter alert patrol problem and the variants thereof, we refer the reader to the authors' prior work (Chandler et al. 2009, Darbha et al. 2010, Krishnamoorthy et al. 2011b,a). Figure 1 shows a typical scenario, where there are 4 alert stations with the UAV at a station (location 0) with an alert. The decision problem we solve is the following: Given that the arrival process of the alerts is Poisson with known arrival rate, what is the optimal time a UAV should spend at a station before resuming its patrol? We associate an information gain with a UAV loitering and servicing an alert and we model this gain as a monotonically increasing function of the loiter/dwell time d .

4.1. Problem Statement

The patrolled perimeter is a simple closed curve with $N(\geq m)$ nodes which are (spatially) uniformly separated, of which m correspond to the alert stations. Let the m distinct station locations be elements of the set $\Omega \subset \{0, \dots, N-1\}$. A typical scenario shown in Figure 1 has 15 nodes, of which, nodes $\{0, 3, 7, 11\}$ correspond to the UGS. Here, station locations 3, 7 and 11 have no alerts, and station location 0 has an alert being serviced by the loitering UAV. At time instant t , let $\ell(t)$ be the position of the UAV on the perimeter ($\ell \in \{0, \dots, N-1\}$), $d(t)$ be the dwell time (number of loiters completed if at an alert site) and $\tau_j(t)$ be the delay in servicing an alert at location $j \in \Omega$. Let $y_j(t)$ be a binary, but random, variable indicating the arrival of an alert at location $j \in \Omega$. We will assume that the statistics associated with the random variable $y_j(t)$ are known and that $y_j; j \in \Omega$ are independent. We model the arrival of alerts as follows: There is a single queue with a Poisson arrival stream of alerts at a rate of α alerts per unit time. After an alert is queued up, we assume it shows up arbitrarily at any one of the m stations (assuming choice of station is a uniformly distributed random variable). For this reason, only one alert can arrive at one of the m stations at any instant of time. Hence, there are $m+1$ possibilities for the value of the vector of alerts $y(t) = [y_1(t) \ y_2(t) \ \dots \ y_m(t)]$, with the first one being that there is no alert at any station and the other m correspond to an alert at each of the m stations. The control decisions are indicated by the variable u . If $u = 1$, then the UAV continues in the same direction as before; if $u = -1$, then the UAV reverses its direction of travel and if $u = 0$, the UAV dwells at the current alert station. We will assume that a UAV advances by one node in unit time if $u \neq 0$. We also assume that the time to complete one loiter is also the unit time. We denote the UAV's direction of travel by ω , where $\omega = 1$ and $\omega = -1$ indicate the clockwise and counter-clockwise directions respectively. One may write the state update equations for the system as follows:

$$\begin{aligned} \ell(t+1) &= [\ell(t) + \omega(t)u(t)] \mod N, \\ \omega(t+1) &= \omega(t)u(t) + \delta(u(t)), \\ d(t+1) &= (d(t) + 1)\delta(u(t)), \end{aligned} \tag{27}$$

$$\tau_j(t+1) = (\tau_j(t) + 1) \{(1 - \delta(\ell(t) - j)\delta(u(t))\} \max\{\sigma(\tau_j(t)), y_j(t)\}, \quad \forall j \in \Omega,$$

where δ is the Kronecker delta function and $\sigma(\cdot) = 1 - \delta(\cdot)$. We denote the status of the alert at station location $j \in \Omega$ at time t by $\mathcal{A}_j(t)$, i.e.,

$$\mathcal{A}_j(t) = \begin{cases} 0, & \text{if } \tau_j(t) = 0 \\ 1, & \text{otherwise} \end{cases}, \quad \forall j \in \Omega. \quad (28)$$

Also, we have the constraints: $u(t) = 0$ only if $\ell(t) \in \Omega$ and $d(t) \leq D$. If $d(t) = D$, then $u(t) \neq 0$ i.e., the UAV is forced to leave the station if it has already completed the maximum (allowed) number of dwell orbits. Combining the different components in (27), we express the evolution equations compactly as:

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), u(t), y(t)),$$

where, $\mathbf{x}(t)$ is the system state at time t with components $\ell(t), \omega(t), d(t)$ and $\tau_j(t), \forall j \in \Omega$. Let us denote the $m+1$ possible values that $y(t)$ can take by the row vector Y_l where,

$$Y_0 = [0 \ 0 \ \dots \ 0], \quad Y_1 = [1 \ 0 \ \dots \ 0], \quad \dots \quad \text{and} \quad Y_m = [0 \ \dots \ 0 \ 1]. \quad (29)$$

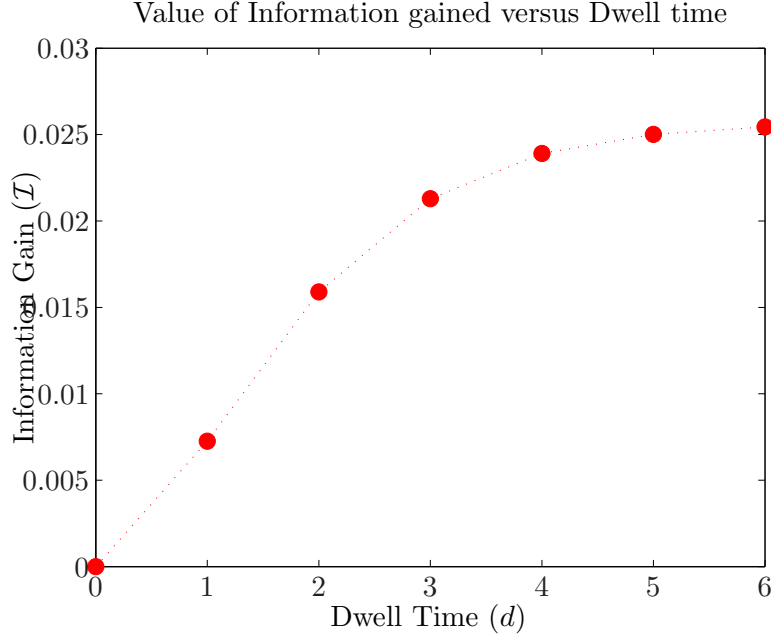
Given a Poisson arrival stream of alerts at the rate of α alerts per unit time, the probability that there is no alert in unit time interval is $p = e^{-\alpha}$ and hence, the probability that $y(t)$ takes any one of the $m+1$ possible values in (29) is given by,

$$p_l := \text{Prob}\{y(t) = Y_l\} = \begin{cases} p, & l = 0, \\ \frac{(1-p)}{m}, & l = 1, \dots, m. \end{cases} \quad (30)$$

To be consistent with the notation introduced earlier (in Sec 2), we shall use \mathcal{S} to denote the set of all system states and use $x \in \{1, \dots, |\mathcal{S}|\}$ to denote a particular state. Our objective is to find a suitable policy that simultaneously minimizes the service delay and maximizes the information gained upon loitering. The information gain, \mathcal{I} , which is based on an operator error model (see Appendix EC.1), is plotted as a function of dwell time in fig. 2. We model the one-step payoff/reward function as follows:

$$R_u(x) = [\mathcal{I}(d_x + 1) - \mathcal{I}(d_x)] \delta(u) - \rho \max\{\bar{\tau}_x, \Gamma\}, \quad x = 1, \dots, |\mathcal{S}|, \quad (31)$$

Figure 2 Value of Information gained vs dwell time.



where d_x is the dwell associated with state x and $\bar{\tau}_x = \max_{j \in \Omega} \tau_{j,x}$ is the worst service delay (among all stations) associated with state x . The parameter $\Gamma (> 0)$ is a judiciously chosen maximum penalty. The positive parameter ρ is a constant weighing the incremental information gained upon loitering once more at the current location against the delay in servicing alerts at other stations. From the state definition, we can compute the total number of states in the MDP to be,

$$|\mathcal{S}| = 2 \times N \times (\Gamma + 1)^m + D \times m \times (\Gamma + 1)^{m-1}, \quad (32)$$

where, the factor 2 comes from the UAV being bi-directional. For the loiter states, directionality is irrelevant and hence when $d \geq 1$, we reset ω to be 1. Note that, in lieu of the reward function definition (31), we do not keep track of delays beyond Γ and hence the state-space \mathcal{S} only includes states x with $\tau_i \leq \Gamma, \forall i \in \Omega$ and so, is finite. We immediately see that the problem size is an m^{th} order polynomial in Γ and hence solving for the optimal value function and policy using exact dynamic programming (DP) methods are rendered intractable for practical values of Γ and m . Hence, we employ the restricted LP approach developed earlier to compute approximate value functions; from which we compute the corresponding greedy sub-optimal policy. In the next section, we exploit the structure in the perimeter patrol problem to simplify the *RLP* and *NLP* formulations and show

that both collapse to exact LPs corresponding to MDPs defined on the M partitions.

4.2. Structure associated with the Perimeter Patrol Problem

In the perimeter patrol problem considered herein, we see that, by definition (31), the reward function $R_u(x)$ is bounded. Consequently the optimal value function is bounded. To explain the inherent structure in the reward, consider a station where an alert is being serviced by a UAV. The information gained by the UAV about the alert is only a function of the service delay at the station and the amount of time the UAV dwells at the station servicing the alert. There is a natural partitioning of states; where no matter what the delays are at the other stations, the reward is the same, as long as the maximum delay and the dwell time of the UAV at the station are the same. So, we aggregate all the states which have the same values for ℓ , ω , d , \mathcal{A}_j , $\forall j \in \Omega$ and $\bar{\tau} = \max_{j \in \Omega} \tau_j$, into one partition. As a result of aggregation, the number of partitions can be shown to be,

$$M = 2 \times N + 2 \times N \times (2^m - 1) \times \Gamma + m \times D + m \times D \times (2^{m-1} - 1) \times \Gamma, \quad (33)$$

which is linear in Γ and hence considerably smaller than the total number of states (32).

We introduce the following notation, that will be used hereafter: Let $\ell_x, d_x, \omega_x, \tau_{j,x}$ and $\mathcal{A}_{j,x}$ represent respectively, the location, dwell, direction of UAV's motion and the service delay and alert status at station location $j \in \Omega$ corresponding to some state $x \in \{1, \dots, |\mathcal{S}|\}$. Also, we will use $\ell(i), d(i), \omega(i), \bar{\tau}(i)$ and $\mathcal{A}_j(i)$ to denote the location, dwell, direction, maximum delay, and the alert status at station location $j \in \Omega$ that correspond to some partition index $i \in \{1, \dots, M\}$. We will also denote by $x(t; x_0, \mathbf{u}_t, \mathbf{y}_t)$ the state at time $t > 0$; if the initial state at $t = 0$ is x_0 and the sequence of inputs, $\mathbf{u}_t = \{u(0), u(1), \dots, u(t-1)\}$ and disturbances, $\mathbf{y}_t = \{y(0), y(1), \dots, y(t-1)\}$. We also introduce a partial ordering of the states according to: $x \geq y$ iff $\ell_x = \ell_y$, $d_x = d_y$, $\omega_x = \omega_y$ and $\tau_{j,x} \geq \tau_{j,y}$, $\forall j \in \Omega$. By the same token, we also partially order partitions, $\mathcal{S}_i \geq \mathcal{S}_j$ iff for every $z \in \mathcal{S}_j$, there exists an $x \in \mathcal{S}_i$ such that $x \geq z$. Recall that $\mathcal{T}(i, u)$ is the set of all distinct $(m+1)$ tuples of partition indices, that the system can transition to, from partition \mathcal{S}_i under control action u . For the sake of notational simplicity, we denote the l^{th} component of any tuple $k \in \mathcal{T}(i, u)$ by

k_{l-1} and the cardinality of the set $\mathcal{T}(i, u)$ by $|\mathcal{T}(i, u)|$. Also we define the partitions to be of two types: a partition \mathcal{S}_i is of type 1 and we write $i \in \mathcal{P}_1$ if $\ell(i) \in \Omega$, $d(i) = 0$, $\mathcal{A}_{\ell(i)}(i) = 1$, and $\mathcal{A}_j(i) = 1$, for some $j \in \Omega, j \neq \ell(i)$, i.e., the UAV is at a station with an alert, the dwell time is zero and also there is an alert at some other station. Else it is of type 2 and we write $i \in \mathcal{P}_2$. Given this definition, we have the following important result, that we will make use of, in the remainder of the paper.

LEMMA 4. *The cardinality of $\mathcal{T}(i, u)$ is given by:*

$$|\mathcal{T}(i, u)| = \begin{cases} \bar{\tau}(i), & i \in \mathcal{P}_1 \text{ and } u = 0, \\ 1, & \text{otherwise.} \end{cases}$$

Proof of Lemma 4. First we consider partition index i of type 1 and control input $u = 0$. Since the UAV has decided to loiter at the current station i.e., $\ell(i) \in \Omega$, the service delay at that station, $\tau_{\ell(i)}$ will be reset to zero in the next time step. Hence the future state (and partition) maximum delay will be determined by the highest of the service delays, say $\bar{\tau}$, among the other stations with alerts (at least one such station exists since partition i is of type 1). So $\forall j \in \{1, \dots, \bar{\tau}(i)\}$, $\exists x_j \in \mathcal{S}_i$ such that $\bar{\tau}_{x_j} = j$. The corresponding tuple of future partition indices $z_{x_j}^{i,0} = (\bar{f}(x_j, u, Y_0), \bar{f}(x_j, u, Y_1), \dots, \bar{f}(x_j, u, Y_m))$ will have maximum delay $j + 1$ and so $\mathcal{T}(i, 0) = \bigcup_{j=1}^{\bar{\tau}(i)} \{z_{x_j}^{i,0}\} \Rightarrow |\mathcal{T}(i, 0)| = \bar{\tau}(i)$. For all other control choices, $u \neq 0$, all the states $x \in \mathcal{S}_i$ will transition to future states with the same maximum delay $\bar{\tau}(i) + 1$. So, for $u \neq 0$, $\mathcal{T}(i, u)$ is a singleton set and hence $|\mathcal{T}(i, u)| = 1$. For partition indices j of type 2 with $\bar{\tau}(j) > 0$, all the states $x \in \mathcal{S}_j$ will transition to future states with the same maximum delay $\bar{\tau}(j) + 1$ and so $|\mathcal{T}(j, u)| = 1, \forall u$. If $\bar{\tau}(j) = 0$, then the partition \mathcal{S}_j is a singleton set as per the aggregation scheme (see Sec 4.2) and hence $|\mathcal{T}(j, u)| = 1, \forall u$. \square

THEOREM 3. *For the perimeter patrol problem, the NLP (26) reduces to the following LP.*

$$\begin{aligned} LBLP &:= \min \bar{c}^T w, \quad \text{subject to} \\ w(i) &\geq r_u(i) + \lambda \sum_{l=0}^m p_l w(k_l), \quad \forall u, i = 1, \dots, M, \end{aligned} \tag{34}$$

where the tuple $k \in \mathcal{T}(i, u)$, if $|\mathcal{T}(i, u)| = 1$, else $k = k^*$, where $k^* \in \mathcal{T}(i, u)$ is the tuple of partition indices such that $\bar{\tau}(k_l^*) = \bar{\tau}(i) + 1, l = 0, \dots, m$. Furthermore, the optimal solution, w^* is dominated by every feasible w for the NLP and, in particular, it is a lower bound to the optimal value function i.e., for all $i = 1, \dots, M$, one has $w^*(i) \leq \min_{x \in \mathcal{S}_i} V^*(x)$.

Before proceeding further, we make two key claims that are essential for the proof of Theorem 3.

The justification for the claims have been provided in the Appendix.

CLAIM 1. If $x_1 \geq x_2$, then for the same sequence of inputs \mathbf{u}_t and disturbances \mathbf{y}_t , the system state evolves in such a way that $x(t; x_1, \mathbf{u}_t, \mathbf{y}_t) \geq x(t; x_2, \mathbf{u}_t, \mathbf{y}_t)$ for every $t > 0$.

CLAIM 2. If $x_1 \geq x_2$, then $V^*(x_1) \leq V^*(x_2)$. Furthermore, if $\mathcal{S}_i \geq \mathcal{S}_j$, then $\min_{x \in \mathcal{S}_i} V^*(x) \leq \min_{z \in \mathcal{S}_j} V^*(z)$.

Proof of Theorem 3. Recall the non-linear constraints (25) satisfied by $\bar{w}(i) := \min_{x \in \mathcal{S}_i} V^*(x)$ that motivated the NLP formulation:

$$\bar{w}(i) \geq \min_{x \in \mathcal{S}_i} \left\{ R_u(x) + \lambda \sum_{l=0}^m p_l \bar{w}(\bar{f}(x, u, Y_l)) \right\}, \quad \forall u, \quad i = 1, \dots, M, \quad (35)$$

which, given the definition of $\mathcal{T}(i, u)$, can be written in the following equivalent form:

$$\bar{w}(i) \geq r_u(i) + \lambda \min_{k \in \mathcal{T}(i, u)} \sum_{l=0}^m p_l \bar{w}(k_l), \quad \forall u, \quad i = 1, \dots, M, \quad (36)$$

where $r_u(i)$ is the reward associated with partition index i , and given the partitioning scheme, satisfies $R_u(x) = r_u(i), \forall x \in \mathcal{S}_i$. Given the structure in the perimeter patrol problem, we will show that the above (36) will collapse to a single linear inequality constraint for every partition index i and control u . Let us focus our attention on partition index i of type 1 and control action $u = 0$. For this choice, the cardinality of $\mathcal{T}(i, 0)$ is $\bar{\tau}(i)$ as per Lemma 4. Indeed $\exists \bar{x} \in \mathcal{S}_i$ such that the corresponding tuple of future partition indices $k^* = (\bar{f}(\bar{x}, 0, Y_0), \bar{f}(\bar{x}, 0, Y_1), \dots, \bar{f}(\bar{x}, 0, Y_m))$ has the highest possible maximum delay, i.e., $\bar{\tau}(k_l^*) = \bar{\tau}(i) + 1, l = 0, \dots, m$. Since $k_l^* \geq k_l, l = 0, \dots, m, \forall k \in \mathcal{T}(i, u)$, we have from Claim 2 that, $\bar{w}(k_l^*) \leq \bar{w}(k_l), l = 0, \dots, m, \forall k \in \mathcal{T}(i, u)$. So, the non-linear inequality corresponding to partition index $i \in \mathcal{P}_1$ and control $u = 0$ becomes:

$$\bar{w}(i) \geq r_0(i) + \lambda \sum_{l=0}^m p_l \bar{w}(k_l^*). \quad (37)$$

If $u \neq 0$, then $|\mathcal{T}(i, u)| = 1$. So there exists exactly one tuple \underline{k} in $\mathcal{T}(i, u)$ and hence, the non-linear constraint (36) reduces to the linear inequality:

$$\bar{w}(i) \geq r_u(i) + \lambda \sum_{l=0}^m p_l \bar{w}(\underline{k}_l). \quad (38)$$

For partition indices j of type 2, $|\mathcal{T}(j, u)| = 1, \forall u$. So, as before, the non-linear inequality (36) collapses to the linear inequality (38).

In summary, we have the following: regardless of which partition one considers, the corresponding non-linear constraint in *NLP* collapses to a linear constraint and hence, *NLP* for the perimeter patrol problem collapses to the following LP:

$$\begin{aligned} LBLP := \min \bar{c}^T w, \quad \text{subject to} \\ w(i) \geq r_u(i) + \lambda \sum_{l=0}^m p_l w(k_l), \quad \forall u, i = 1, \dots, M, \end{aligned} \quad (39)$$

where the tuple $k \in \mathcal{T}(i, u)$, if $|\mathcal{T}(i, u)| = 1$, else $k = k^*$, where $k^* \in \mathcal{T}(i, u)$ is the tuple of partition indices such that $\bar{\tau}(k_l^*) = \bar{\tau}(i) + 1, l = 0, \dots, m$.

To prove the second part of the Theorem, we observe that *LBLP* defined above is the exact LP corresponding to a reduced order MDP defined on the M partitions. Hence, we readily have from Lemmas 1 and 2 that the optimal solution w^* lower bounds every feasible solution including \bar{w} and hence, $w^*(i) \leq \bar{w}(i) = \min_{x \in \mathcal{S}_i} V^*(x) \leq V^*(y), \forall y \in \mathcal{S}_i, i = 1, \dots, M$. \square

So, for the perimeter patrol problem, one can compute a lower bound for the optimal value function efficiently by solving *LBLP*. The next logical question is whether the upper bound formulation, *RLP* (13), also simplifies, given the structure in the problem. It turns out that this is indeed the case, as can be seen from the following theorem.

THEOREM 4. *For the perimeter patrol problem, the RLP (13) reduces to the following LP.*

$$\begin{aligned} UBLP := \min \bar{c}^T w, \quad \text{subject to} \\ w(i) \geq r_u(i) + \lambda \sum_{l=0}^m p_l w(k_l), \quad \forall u, i = 1, \dots, M, \end{aligned} \quad (40)$$

where the tuple $k \in \mathcal{T}(i, u)$, if $|\mathcal{T}(i, u)| = 1$, else $k = k^*$, where $k^* \in \mathcal{T}(i, u)$ is the tuple of partition indices such that $\bar{\tau}(k_l^*) = 2, l = 0, \dots, m$.

Proof of Theorem 4. Given the partitioning scheme, one can rewrite the Bellman inequalities (5) as follows: for each $i = 1, \dots, M$,

$$V^*(x) \geq r_u(i) + \lambda \sum_{l=0}^m p_l V^*(f(x, u, Y_l)), \forall u, \forall x \in \mathcal{S}_i. \quad (41)$$

With the restriction that $V(x) = v(i), \forall x \in \mathcal{S}_i$, we get the following constraint for *RLP* (13),

$$v(i) \geq r_u(i) + \lambda \sum_{l=0}^m p_l v(k_l), \forall k \in \mathcal{T}(i, u), \forall u, i = 1, \dots, M. \quad (42)$$

For partition index $i \in \mathcal{P}_1$, $\exists \bar{x} \in \mathcal{S}_i$ that transitions to future states with the least possible maximum delay, 2. Hence $f(\bar{x}, 0, Y_l) \leq f(x, 0, Y_l)$, $l = 0, \dots, m$, $\forall x \in \mathcal{S}_i$ and so from Claim 2 we have, $V^*(f(\bar{x}, 0, Y_l)) \geq V^*(f(x, 0, Y_l))$, $l = 0, \dots, m$, $\forall x \in \mathcal{S}_i$. So, for $i \in \mathcal{P}_1$ and $u = 0$, the inequalities (41) can be written as follows,

$$V^*(x) \geq r_0(i) + \lambda \sum_{l=0}^m p_l V^*(f(\bar{x}, 0, Y_l)) \forall u, \forall x \in \mathcal{S}_i. \quad (43)$$

The above implies that the $\bar{\tau}(i)$ constraints (42) in *RLP* can be replaced by the single constraint,

$$v(i) \geq r_0(i) + \lambda \sum_{l=0}^m p_l v(k_l^*), \quad (44)$$

where $k^* = (\bar{f}(\bar{x}, 0, Y_0), \bar{f}(\bar{x}, 0, Y_1), \dots, \bar{f}(\bar{x}, 0, Y_m))$ is the tuple of future partition indices (corresponding to \bar{x}) with the least possible maximum delay, i.e., $\bar{\tau}(k_l^*) = 2$, $l = 1, \dots, m$. For the other control choices, $u \neq 0$, there exists only one tuple \bar{k} in $\mathcal{T}(i, u)$ (since $|\mathcal{T}(i, u)| = 1$) and hence the constraint (42) is the single constraint,

$$v(i) \geq r_u(i) + \lambda \sum_{l=0}^m p_l v(\bar{k}_l), u \neq 0. \quad (45)$$

Similarly, for partitions \mathcal{S}_j of type 2, $|\mathcal{T}(j, u)| = 1$, $\forall u$, and so the constraint (42) is the single constraint (45).

In summary, we have the following: regardless of which partition index $i \in \{1, \dots, M\}$ and control action u are considered, the corresponding $|\mathcal{T}(i, u)|$ linear constraints in *RLP* collapse to a single constraint and hence, *RLP* for the perimeter patrol problem reduces to the following exact LP:

$$UBLP := \min \bar{c}^T w, \quad \text{subject to}$$

$$w(i) \geq r_u(i) + \lambda \sum_{l=0}^m p_l w(k_l), \quad \forall u, \quad i = 1, \dots, M, \quad (46)$$

where the tuple $k \in \mathcal{T}(i, u)$, if $|\mathcal{T}(i, u)| = 1$, else $k = k^*$, where $k^* \in \mathcal{T}(i, u)$ is the tuple of partition indices such that $\bar{\tau}(k_l^*) = 2$, $l = 0, \dots, m$. \square

In conclusion, we have two complementary LP formulations, *UBLP* and *LBLP* that can be used to efficiently compute upper bound and lower bound approximate value functions respectively, for the perimeter alert patrol problem. Note that the two formulations involve computing the optimal value functions for reduced order MDPs defined over the M partitions and in that sense are computationally attractive (compared to solving the original problem) since $M \ll |\mathcal{S}|$. In the following section, we will provide numerical results that corroborate the key claims made earlier regarding the structure in the perimeter alert patrol problem.

5. Numerical Results

We consider a perimeter with $N = 15$ nodes of which node numbers $\{0, 3, 7, 11\}$ are alert stations and a maximum allowed dwell of $D = 5$ orbits. The other parameters were chosen to be: weighing factor, $\rho = .005$ and temporal discount factor, $\lambda = 0.9$. Based on experience, we chose the alert arrival rate $\alpha = \frac{1}{60}$. This reflects a rather low arrival rate where we expect 2 alerts to occur on average in the time taken by the UAV to complete an uninterrupted patrol around the perimeter. We set the maximum delay time, that we keep track of, to be $\Gamma = 15$; for which the total number of states comes out to be $|\mathcal{S}| = 2,048,000$. Before venturing into the simulation, we first provide numerical results that corroborate the key Claim 2, made earlier in the paper. For this, we solve for the optimal value function V^* . This is possible since the size of the example problem considered in this section is small and hence an exact solution can be obtained. In Figure 3, we show results supporting the claim that for partially ordered states $x_1 \geq x_2$, the corresponding optimal value functions satisfy $V^*(x_1) \leq V^*(x_2)$. For this, we plot the optimal value function V^* corresponding to states with alert status $\mathcal{A}_j = 1, \forall j \in \Omega$ (all stations have alerts), dwell $d = 0$, direction $\omega = 1$ and the UAV located at one of the four station locations $\ell \in \Omega$. The partially ordered states represented

Figure 3 Monotonically decreasing value function corresponding to partially ordered states with increasing maximum delay.

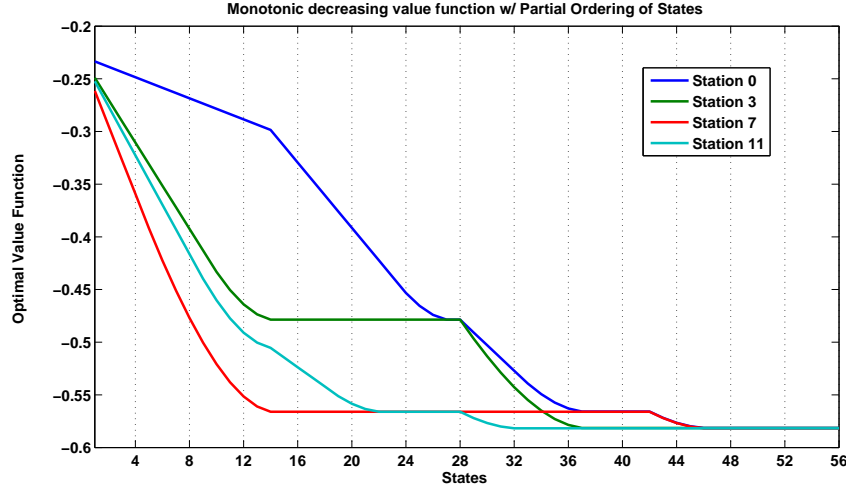
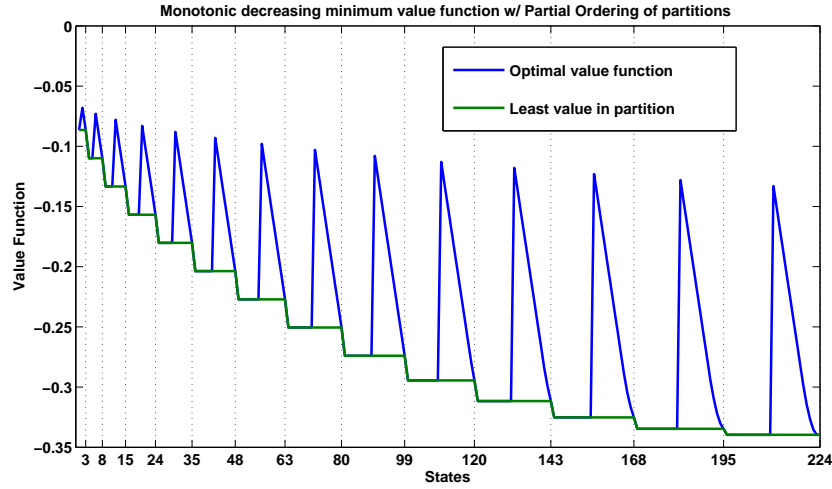


Figure 4 Monotonically decreasing least value function corresponding to partially ordered partitions with increasing maximum delay.



in the X-axis are non-decreasing from left to right with maximum delay $\bar{\tau}$ varying from 2 to Γ . The dotted grid lines in the plot separate the different partitions that the states fall into. In Figure 4, we show results supporting the claim that for partially ordered partitions $\mathcal{S}_i \geq \mathcal{S}_j$, the corresponding optimal value functions satisfy $\min_{x \in \mathcal{S}_i} V^*(x) \leq \min_{y \in \mathcal{S}_j} V^*(y)$. For this, we plot the value functions corresponding to states with alert status $\mathcal{A} = 1001$ (station locations 0 and 11 have alerts), dwell $d = 0$, direction $\omega = 1$ and $\ell = 0$. The partially ordered partitions demarcated by the dotted grid

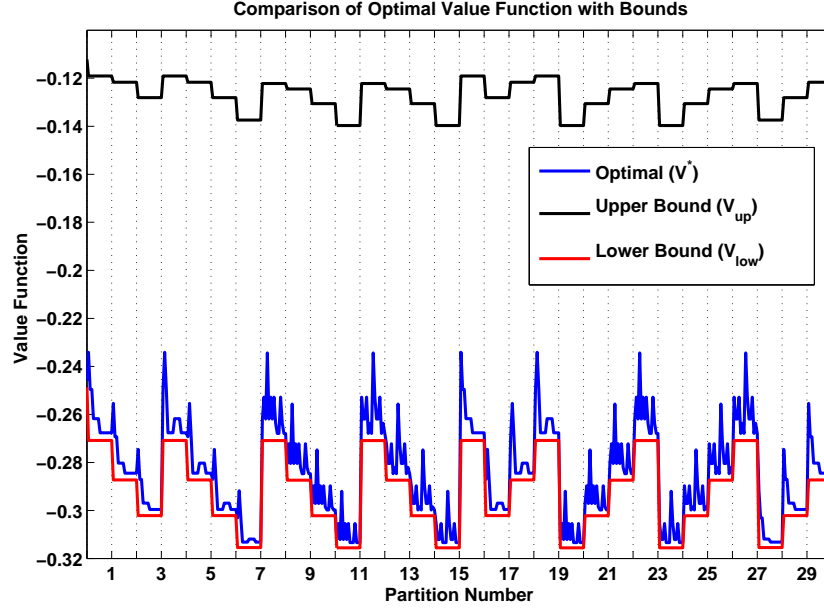
lines in the X-axis are non-decreasing from left to right with maximum delay $\bar{\tau}$ varying from 2 to Γ . Within each partition, we plot the value function associated with every state in the partition and also the least value function in the partition shown as the green line. One can easily see that the claim above is satisfied.

In the next section, we shall consider the same example problem and show that the proposed approximate methodology is effective. For this, we compute the approximate value functions via the restricted LP formulation and compare them with the optimal value function. In addition, we also compute the greedy sub-optimal policy corresponding to the approximate value function and compare it with the optimal policy in terms of the two performance metrics: alert service delay and information gained upon loitering.

5.1. Simulation Results

We aggregate the states in the example problem based on the reward function (see section 4.2 for details). This results in $M = 8900$ partitions, which is considerably smaller than the original number of states, $|S|$. We solve both the *UBLP* and *LBLP* formulations which give us the upper and lower bounds, v^* and w^* respectively, to the optimal value function V^* . Since we have the optimal value function for the example problem, we use it for comparison with the approximations. Note that for higher values of m and Γ , the problem would essentially become intractable and one would not have access to the optimal value function. Nevertheless, one can compute v^* and w^* and the difference between the two would give an estimate of the quality of the approximation. We give a representative sample of the approximation results by choosing all the states in partitions corresponding to alert status $\mathcal{A}_j = 1, \forall j \in \Omega$ (all stations have alerts) and maximum delay $\bar{\tau} = 2$. Figure 5 compares the optimal value function V^* with the upper and lower bound approximate value functions, $V_{up} = \Phi v^*$ and $V_{low} = \Phi w^*$ for this subset of the state-space. The first 15 partitions shown in the X-axis of Figure 5 i.e., partition numbers, $i = 1, \dots, 15$, correspond to the clockwise states:

$$\ell = i - 1, \quad d = 0, \quad \omega = 1, \quad \bar{\tau} = \max_{j \in \Omega} \tau_j = 2, \quad \mathcal{A}_j = 1, \quad \forall j \in \Omega, \quad (47)$$

Figure 5 Comparison of approximate value functions with the optimal.

and the last 15 partitions shown in the X-axis i.e., partition numbers, $i = 16, \dots, 30$, correspond to the counter-clockwise states:

$$\ell = i - N - 1, \quad d = 0, \quad \omega = -1, \quad \bar{\tau} = \max_{j \in \Omega} \tau_j = 2, \quad \mathcal{A}_j = 1, \quad \forall j \in \Omega. \quad (48)$$

Interestingly, we notice immediately that the lower bound appears to be tighter than the upper bound. Recall that our objective is to obtain a good sub-optimal policy and so, we consider the policy that is *greedy* with respect to V_{low} :

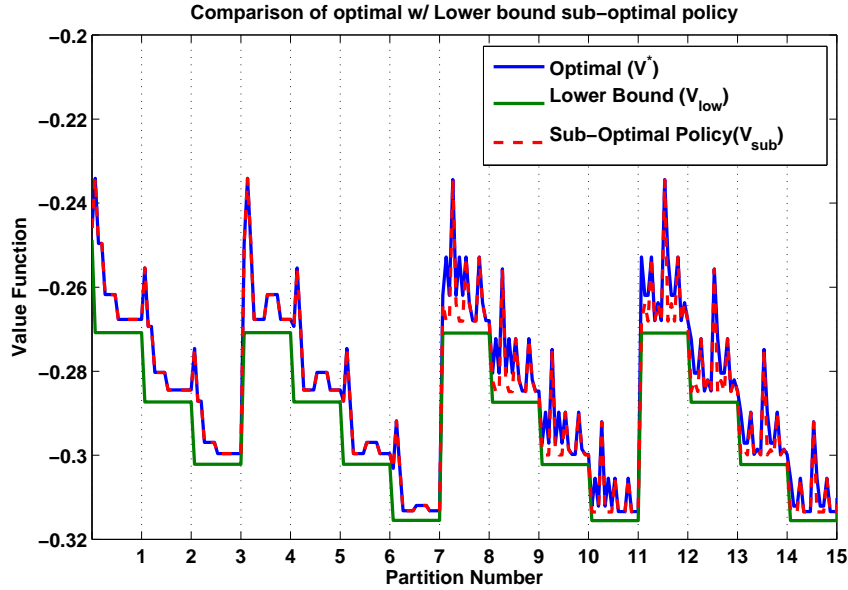
$$\pi_s(x) = \arg \max_u \left\{ R_u(x) + \lambda \sum_{l=0}^m p_l V_{low}(f(x, u, Y_l)) \right\}, \quad \forall x \in \{1, \dots, |\mathcal{S}|\}. \quad (49)$$

To assess the quality of the sub-optimal policy, we also compute the expected discounted payoff, V_{sub} that corresponds to the sub-optimal policy π_s , by solving the system of equations:

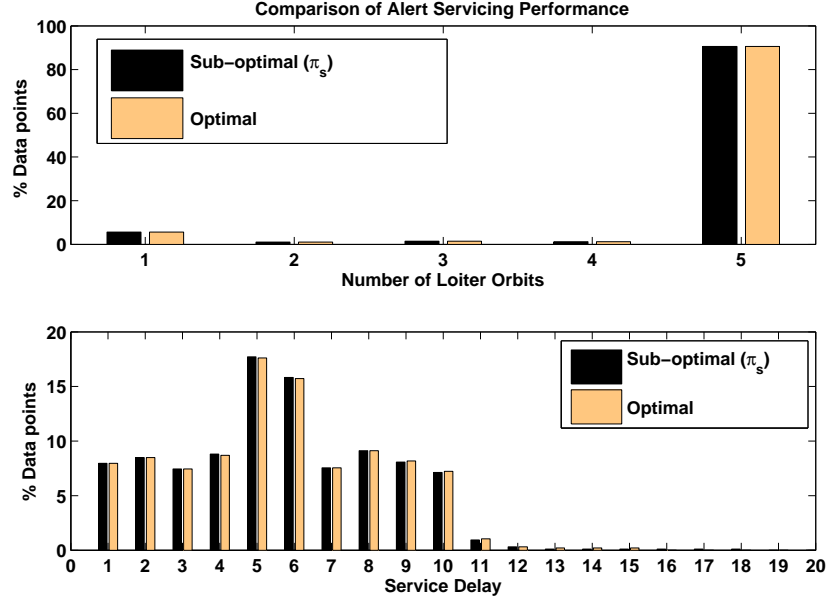
$$(I - \lambda P_{\pi_s}) V_{sub} = R_{\pi_s}. \quad (50)$$

Since V_{sub} corresponds to a sub-optimal policy and in lieu of the monotonicity property of the Bellman operator, the following inequalities hold:

$$V_{low} \leq V_{sub} \leq V^* \leq V_{up}.$$

Figure 6 Comparison of value function corresponding to suboptimal policy π_s with the optimal.

In Figure 6, we compare V_{sub} with the optimal value function V^* for the clockwise states defined in (47) and note that the approximation is quite good. Finally, we compare the performance of the sub-optimal policy π_s with that of the optimal strategy π^* in terms of the two important metrics: service delay and information gain (measured via the dwell time). To collect the performance statistics, we ran Monte Carlo simulations with alerts generated from a Poisson arrival stream with rate $\alpha = \frac{1}{60}$ over a 60000 time unit simulation window. Both the optimal and sub-optimal policies were tested against the same alert sequence. Figure 7 shows histogram plots for the service delay (top plot) and the dwell time (bottom plot) for all serviced alerts in the simulation run. The corresponding mean and worst case service delays and the mean dwell time are also shown in Table 1. We see that there is hardly any difference in terms of either metric between the optimal and the sub-optimal policies. This substantiates the claim that the aggregation approach gives us a sub-optimal policy that performs almost as well as the optimal policy itself. This is to be expected, given that the value functions corresponding to the optimal and sub-optimal policies are close to each other (see Figure 6). Since the false alarm rate α is fairly low, we see from the bottom plot of Figure 7 that roughly 90% of the alerts were cleared within ten time steps. Also from the top plot of Figure 7, we see that maximum information was gained (5 loiters completed) on almost 90% of

Figure 7 Comparison of service delay and number of loiters between optimal and sub-optimal policies.**Table 1** Comparison of alert servicing performance between optimal and sub-optimal policies.

Policy	Mean number of loiters	Mean service delay	Worst service delay
π^*	4.7	5.6	15
π_s	4.7	5.6	18

the serviced alerts.

6. Conclusions

We have provided a state aggregation based restricted LP method to construct sub-optimal policies for stochastic DPs along with a bound for the deviation of such a policy from the optimum value function. As a key result, we have shown that the solution to the aggregation based LP is independent of the underlying cost function and we do so by demonstrating that the restricted LP is, in fact, the exact LP that corresponds to a lower dimensional MDP defined over the partitions. We also provide a novel non-linear program that can be used to compute a non-trivial lower bound to the optimal value function. In particular, for the perimeter patrol stochastic control problem, we have shown that both the upper and lower bound formulations simplify to exact LPs corresponding to some reduced order MDPs. To do so, we have exploited the partial ordering of the states that

comes about because of the structure inherent in the reward function. It would be interesting to see if the simplification can be achieved for other problems that exhibit a similar structure. For the perimeter patrol problem, numerical results obtained via Monte Carlo simulations show that the sub-optimal policy obtained via the approximate value functions perform almost as well as the optimal policy. The literature suggests that, in general, the solution to a restricted LP depends on the underlying cost function; when the value function is parameterized by arbitrary basis functions. We have shown that, for the special case of hard aggregation, this is not true. Surely, there exist other basis functions with the same property and it would be useful to uncover the class of basis functions, for which the independence result holds.

References

- Axsäter, S. 1983. State aggregation in dynamic programming: An application to scheduling of independent jobs on parallel processors. *Oper. Res. Letters* **2** 171–176.
- Balas, E. 1979. *Disjunctive programming*, *Annals of Discrete Mathematics*, vol. 5. North-Holland Publishing Company, 3–51.
- Balas, E. 1998. Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Applied Math.* **89**(1-3) 3–44.
- Bean, J. C., J. R. Birge, R. L. Smith. 1987. Aggregation in dynamic programming. *Oper. Res.* **35** 215–220.
- Bellman, R. E. 1957. *Dynamic Programming*. Princeton University Press, Princeton, NJ.
- Bertsekas, D. P. 2007. *Dynamic Programming and Optimal Control*, vol. II, chap. Approximate Dynamic Programming. 3rd ed. Athena Scientific.
- Chandler, P., J. Hansen, R. Holsapple, S. Darbha, M. Pachter. 2009. Optimal perimeter patrol alert servicing with Poisson arrival rate. *AIAA Guidance, Navigation and Control Conf.*. Chicago, IL.
- Darbha, S., K. Krishnamoorthy, M. Pachter, P. Chandler. 2010. State aggregation based linear programming approach to approximate dynamic programming. *Proc. IEEE Conf. Decision and Control*. Atlanta, GA, 935–941.
- De Farias, D. P., B. Van Roy. 2003. The linear programming approach to approximate dynamic programming. *Oper. Res.* 850–865.

- Denardo, E. V. 1970. On linear programming in a Markov decision problem. *Management Sci.* **16**(5) 282–288.
- d’Epenoux, F. 1963. A probabilistic production and inventory problem. *Management Sci.* **10**(1) 98–108.
- Glover, F. 1968. Surrogate constraints. *Oper. Res.* **16**(4) 741–749.
- Glover, F. 1975. Surrogate constraint duality in mathematical programming. *Oper. Res.* **23**(3) 434–451.
- Gordon, G. 1999. Approximate solutions to Markov decision processes. Ph.D. thesis, Carnegie Mellon University, Pittsburg, PA.
- Greenberg, H. J., W. P. Pierskalla. 1970. Surrogate mathematical programming. *Oper. Res.* **18**(5) 924–939.
- Gross, D., S. Rasmussen, P. Chandler, G. Feitshans. 2006. Cooperative Operations in Urban TERRain (COUNTER). *Defense and Security Sympos.* SPIE, Orlando, FL.
- Grötschel, M., O. Holland. 1991. Solution of large-scale symmetric travelling salesman problems. *Math. Programming* **51** 141–202.
- Grötschel, M., L. Lovász, A. Schijver. 1981. The ellipsoid method and its consequences in combinatorial optimization. *combinatorica* **1**(2) 169–197.
- Hordijk, A., L. C. M. Kallenberg. 1979. Linear programming and Markov decision chains. *Management Sci.* **25**(4) 352–362.
- Howard, R. A. 1960. *Dynamic Programming and Markov Processes*. The MIT Press, Cambridge, MA.
- Krishnamoorthy, K., M. Pachter, P. Chandler, D. Casbeer, S. Darbha. 2011a. UAV perimeter patrol operations optimization using efficient dynamic programming. *American Control Conf.* San Francisco, CA.
- Krishnamoorthy, K., M. Pachter, S. Darbha, P. Chandler. 2011b. Approximate dynamic programming with state aggregation applied to UAV perimeter patrol. *Internat. J. of Robust and Nonlinear Control* **21**.
- Manne, A. S. 1960. Linear programming and sequential decisions. *Management Sci.* **6**(3) 259–267.
- Mendelssohn, R. 1980. Improved bounds for aggregated linear programs. *Oper. Res.* **28**(6) 1450–1453.
- Mendelssohn, R. 1982. An iterative aggregation procedure for Markov decision processes. *Oper. Res.* **30**(1) 62–73.
- Morrison, J. R., P. R. Kumar. 1999. New linear program performance bounds for queueing networks. *J. Optim. Theory and Appl.* **100**(3) 575–597.

- Porteus, E. L. 1975. Bounds and transformations for discounted finite Markov decision chains. *Oper. Res.* **23**(4) 761–784.
- Schuurmans, D., R. Patrascu. 2001. *Direct value-approximation for factored MDPs*, *Advances in Neural Information Processing Systems*, vol. 14. MIT Press, Cambridge, MA, 1579–1586.
- Schweitzer, P. J., A. Seidmann. 1985. Generalized polynomial approximations in Markovian decision processes. *J. Math. Anal. and Appl.* **110**(2) 568–582.
- Singh, S. P., T. Jaakkola, M. I. Jordan. 1995. Reinforcement learning with soft state aggregation. *Advances in Neural Information Processing Systems 7: Proceedings of the 1994 Conference*. MIT Press, 361–368.
- Trick, M., S. Zin. 1993. A linear programming approach to solving stochastic dynamic programs.
- Trick, M., S. Zin. 1997. Spline approximation to value functions: A linear programming approach. *Macroeconomic Dynamics* **1** 255–277.
- Van Roy, B. 2006. Performance loss bounds for approximate value iteration with state aggregation. *Math. Oper. Res.* **31**(2) 234–244.
- Wang, Y., S. Boyd. 2010. Approximate dynamic programming via iterated Bellman inequalities. URL http://www.stanford.edu/~boyd/papers/adp_iter_bellman.html.

This page is intentionally blank. Proper e-companion title page, with INFORMS branding and exact metadata of the main paper, will be produced by the INFORMS office when the issue is being assembled.

Appendix to “Bounding Procedures for Stochastic Dynamic Programs with Application to the Perimeter Alert Patrol Problem” by Park et al.

This appendix contains supplementary material to the paper and also lengthy proofs that were left out of the main document.

EC.1. Operator Error Model

We treat the operator as a sensor-in-the-loop automaton. The operator is not infallible and we account for that statistically in the optimization. To quantify the operator’s performance, we consider two random variables: the variable X that specifies whether the alert is a real threat (target T) or a nuisance (false target FT) and the operator decision Z which specifies whether he determines the alert to be a real threat Z_1 or a nuisance Z_2 . We stipulate that the a priori probability that an alert is a real target,

$$Prob\{X = T\} = p \ll 1. \quad (\text{EC.1})$$

We assume, based on experience, that $p = 0.01$ in this work. The conditional probabilities which specify whether the operator correctly reported a threat and a nuisance are assumed to be functions of the dwell time, d :

$$\begin{aligned} P_{TR}(d) &:= Prob\{Z = Z_1 | X = T\} = a + b(1 - e^{-\mu_1 d}), \\ P_{FTR}(d) &:= Prob\{Z = Z_2 | X = FT\} = c + g(1 - e^{-\mu_2 d}). \end{aligned} \quad (\text{EC.2})$$

where the acronyms TR and FTR stand for *Target Report* and *False Target Report* respectively.

The parameters a, b, μ_1, c, g, μ_2 characterize the “confusion matrix” and the performance of the operator as a sensor; for details on sensor performance modeling, see Sec 7.2 in Kish et al. (2009).

The parameters satisfy the constraints:

$$0 < a + b \leq 1, \quad 0 < c + g \leq 1, \quad \mu_1 \geq 0 \quad \text{and} \quad \mu_2 \geq 0.$$

In this work, we chose $a = c = 0.5$, $b = g = 0.45$ and $\mu_1 = \mu_2 = 1$. The choice $a = c = 0.5$ correspond to an uninformed or unbiased operator, i.e., the operator cannot tell if the alert is a threat or a

nuisance without having seen any video footage of the alert site. We wish to maximize the mutual information - derived along the lines of information theory (Cover and Thomas 2006) - between the random variables X and Z given by:

$$\begin{aligned}\mathcal{I}(X; Z) &= H(X) - H(X|Z) \\ &= \sum_{x,z} \text{Prob}\{X=x, Z=z\} \log \frac{\text{Prob}\{X=x, Z=z\}}{\text{Prob}\{X=x\}\text{Prob}\{Z=z\}},\end{aligned}\quad (\text{EC.3})$$

where $H(X)$ is the entropy of X and $H(X|Z)$ is the conditional entropy of X given Z . Using Bayes' rule and the probabilities (EC.1) and (EC.2), one can show that the mutual information is a function of dwell time, d :

$$\begin{aligned}\mathcal{I}(d) &= pP_{TR} \log \frac{P_{TR}}{pP_{TR} + (1-p)(1-P_{FTR})} \\ &\quad + p(1-P_{TR}) \log \frac{1-P_{TR}}{p(1-P_{TR}) + (1-p)P_{FTR}} \\ &\quad + (1-p)(1-P_{FTR}) \log \frac{1-P_{FTR}}{pP_{TR} + (1-p)(1-P_{FTR})} \\ &\quad + (1-p)P_{FTR} \log \frac{P_{FTR}}{p(1-P_{TR}) + (1-p)P_{FTR}},\end{aligned}\quad (\text{EC.4})$$

since the conditional probabilities, P_{TR} and P_{FTR} are both functions of d (EC.2).

EC.2. Proofs to lemma in Section 2.1

LEMMA 1. *Let the vector V satisfy the following set of inequalities:*

$$[I - \lambda^L P_\pi^L] V \geq [I + \lambda P_\pi + \dots + \lambda^{L-1} P_\pi^{L-1}] R_\pi, \quad \forall \pi. \quad (\text{EC.5})$$

Then, we have $V \geq V^$.*

Proof of Lemma 1. For every stationary policy π , we have:

$$[I - \lambda^L P_\pi^L] V \geq [I + \lambda P_\pi + \dots + \lambda^{L-1} P_\pi^{L-1}] R_\pi. \quad (\text{EC.6})$$

Since P_π is a stochastic matrix (i.e., it is non-negative and its row sum equals 1), and $\lambda \in [0, 1)$, the matrix $[I - \lambda^L P_\pi^L]^{-1}$ admits the following analytic series expansion:

$$[I - \lambda^L P_\pi^L]^{-1} = I + \lambda^L P_\pi^L + \lambda^{2L} P_\pi^{2L} + \dots$$

So, all the entries of $[I - \lambda^L P_\pi^L]^{-1}$ are non-negative and hence (EC.6) implies the following (although the converse is not true!):

$$V \geq [I - \lambda^L P_\pi^L]^{-1} [I + \lambda P_\pi + \cdots + \lambda^{L-1} P_\pi^{L-1}] R_\pi = \sum_{i=0}^{\infty} \lambda^i P_\pi^i R_\pi, \quad \forall \pi. \quad (\text{EC.7})$$

So, V dominates the expected payoff associated with every policy π , including the optimal policy π^* . Hence $V \geq V^*$. \square

EC.3. Proof to lemma in Section 3.1

LEMMA 3. Consider a surrogate LP for the RLP through a set of dual variables, μ given by:

$$\begin{aligned} SLP(\mu) &:= \min \bar{c}^T v, \quad \text{subject to} \\ \sum_{x \in \mathcal{S}_i} \mu_u^i(x) v(i) &\geq \sum_{x \in \mathcal{S}_i} \mu_u^i(x) \left[R_u(x) + \lambda \sum_{l=0}^m p_l v(\bar{f}(x, u, Y_l)) \right], \quad \forall u, i = 1, \dots, M. \end{aligned} \quad (\text{EC.8})$$

Then, $\exists \bar{\mu} \geq 0$ such that, $SLP(\bar{\mu}) = RLP$, and, for every partition index $i = 1, \dots, M$, $\exists u$ such that $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. Moreover, the optimal solution v^* to RLP is independent of the cost vector \bar{c} and any other feasible solution v to RLP dominates v^* .

Proof of Lemma 3. Consider the Langrangian dual problem to RLP,

$$LD(\mu) := \min_v \left\{ \bar{c}^T v - \sum_{i,u} \sum_{x \in \mathcal{S}_i} \mu_u^i(x) \left[v(i) - R_u(x) - \lambda \sum_{l=0}^m p_l v(\bar{f}(x, u, Y_l)) \right] \right\}.$$

Let $\phi(v, \mu) = \bar{c}^T v - \sum_{i,u} \sum_{x \in \mathcal{S}_i} \mu_u^i(x) [v(i) - R_u(x) - \lambda \sum_{l=0}^m p_l v(\bar{f}(x, u, Y_l))]$. Let \mathcal{F} be the feasible set for RLP and let $\mathcal{F}(\mu)$ be the feasible set of $SLP(\mu)$. Then, we have,

$$\begin{aligned} LD(\mu) &:= \min_v \phi(v, \mu) \\ &\leq \min_{v \in SLP(\mu)} \phi(v, \mu) \\ &\leq \min_{v \in SLP(\mu)} \bar{c}^T v = SLP(\mu). \end{aligned}$$

Since $\mathcal{F} \subset \mathcal{F}(\mu)$ for every μ , it readily follows that $SLP(\mu) \leq RLP$. Also, RLP is feasible. For eg., consider the feasible solution \tilde{v} given by,

$$\tilde{v}(i) = \frac{\max_{x,u} R_u(x)}{1 - \lambda}, \quad \forall i \in \{1, \dots, M\}.$$

Moreover, any feasible v satisfies,

$$v(i) \geq \frac{\min_{x,u} R_u(x)}{1-\lambda}, \forall i \in \{1, \dots, M\}.$$

So, RLP is also bounded from below and hence it satisfies the requirements of strong duality for LPs. Hence, there exists a $\bar{\mu}$ which is optimal for the dual of RLP and also satisfies $LD(\bar{\mu}) = RLP$. Therefore, the same $\bar{\mu}$ must also be such that $SLP(\bar{\mu}) = RLP$. Now for every partition index $i = 1, \dots, M$, there exists at least one u for which $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. If for some i , $\bar{\mu}_u^i(x) = 0$ for every $x \in \mathcal{S}_i$ and for every u , then $SLP(\bar{\mu})$ will not have any constraints lower bounding $v(i)$. It will then admit solutions for $v(i)$ that are arbitrarily negative and correspondingly, one can find a direction in which the cost of $SLP(\bar{\mu})$ decreases without bound. However, this is a contradiction, since RLP is lower bounded. So, we can rewrite $SLP(\bar{\mu})$ in the following manner:

$$SLP(\bar{\mu}) := \min \bar{c}^T v, \quad \text{subject to} \tag{EC.9}$$

$$v(i) \geq r_u(i) + \lambda \frac{1}{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x)} \sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) \sum_{l=0}^m p_l v(\bar{f}(x, u, Y_l)), \quad \forall u \in \mathcal{U}_i, i = 1, \dots, M,$$

where, $u \in \mathcal{U}_i$ if $\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) > 0$. Clearly, $SLP(\bar{\mu})$ is the exact LP corresponding to a MDP of reduced dimension with one-step reward function,

$$r_u(i) = \frac{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) R_u(x)}{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x)}, \quad \forall u \in \mathcal{U}_i,$$

and transition probability matrix \tilde{P}_u given by,

$$\tilde{P}_u(i, j) := \begin{cases} \frac{1}{\sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x)} \sum_{x \in \mathcal{S}_i} \bar{\mu}_u^i(x) \sum_{y \in \mathcal{S}_j} P_u(x, y), & \text{if } u \in \mathcal{U}_i, \\ 0, & \text{otherwise.} \end{cases}$$

So, by Lemma 2, the optimal solution v^* is also the optimal value function associated with the same underlying MDP. Also any feasible v to RLP is also a feasible solution to $SLP(\bar{\mu})$ since the constraints for $SLP(\bar{\mu})$ are obtained by a convex combination of the constraints of RLP . So, it follows from Lemma 1 that $v \geq v^*$.

Finally, let $RLP(\bar{c})$ and $RLP(\bar{d})$ denote the restricted LPs corresponding to two different cost vectors \bar{c} and \bar{d} respectively. Let the corresponding optimal solutions be v_c^* and v_d^* . Since v_d^* is a feasible solution for $RLP(\bar{c})$, we have $v_d^* \geq v_c^*$. By the same token, $v_c^* \geq v_d^*$. Hence, $v_c^* = v_d^*$. \square

EC.4. Proofs to claims in Section 4.2

CLAIM 1. *If $x_1 \geq x_2$, then for the same sequence of inputs \mathbf{u}_t and disturbances \mathbf{y}_t , the system state evolves in such a way that $x(t; x_1, \mathbf{u}_t, \mathbf{y}_t) \geq x(t; x_2, \mathbf{u}_t, \mathbf{y}_t)$ for every $t > 0$.*

Proof of Claim 1. We use induction. Clearly at $t = 0$, $x_1 \geq x_2$. By the semi-group property of state transitions, it is sufficient to show that the result holds for $t = 1$. We define the state, x , of the patrol system to be of two types. If the following holds:

$$\ell_x \in \Omega, d_x = 0, \mathcal{A}_{\ell_x, x} = 1, \text{ and } \mathcal{A}_{j, x} = 1, \text{ for some } j \in \Omega, j \neq \ell_x, \quad (\text{EC.10})$$

i.e., the UAV is at a station with an alert, the dwell time is zero and also there is an alert at some other station, then the state x is of type 1. Else it is of type 2. Note that if $x_1 \geq x_2$, then the states x_1 and x_2 are necessarily of the same type. The key property we will be using in proving Claim 1 is the following: service delay at a station either remains at zero (if no new alert has occurred there) or it goes up by 1 (if there is an unserved alert there) or it is reset to zero (if a UAV decides to loiter there).

If x_1 and x_2 are of type 1 and the UAV chooses to loiter, i.e., $u(0) = 0$, we clearly see that neither the location nor the dwell will differ at $t = 1$. Furthermore, the delays at $t = 1$ associated with the stations corresponding to initial state x_1 will be no less than the delays associated with stations corresponding to initial state x_2 since $x_1 \geq x_2$. If $z_1 = x(1; x_1, 0, y(0))$ and $z_2 = x(1; x_2, 0, y(0))$, we see that $\ell_{z_1} = \ell_{z_2}$, $d_{z_1} = d_{z_2}$, $\omega_{z_1} = \omega_{z_2}$, and $\tau_{j, z_1} \geq \tau_{j, z_2}$, $\forall j \in \Omega$ for every disturbance $y(0)$ and so $z_1 \geq z_2$. The same relationship holds for other possible control choices, $u(0) \neq 0$, as well. By a similar argument, one can show that $x(1; x_1, u(0), y(0)) \geq x(1; x_2, u(0), y(0))$ holds, regardless of the control choice, even if the states x_1, x_2 are of type 2. We use the semi-group property as follows: suppose the claim holds for all t lying between 0 and l for some $l > 0$. Then, we will treat the state at $t = l$ as the initial condition for determining the evolution of the state at $t = l + 1$. The clock is reset as: $\tilde{t} = t - l$, $t \geq l$. By the preceding arguments, Claim 1 holds for $\tilde{t} = 1$ which is equivalent to saying that it holds for $t = l + 1$. \square

CLAIM 2. If $x_1 \geq x_2$, then $V^*(x_1) \leq V^*(x_2)$. Furthermore, if $\mathcal{S}_i \geq \mathcal{S}_j$, then $\min_{x \in \mathcal{S}_i} V^*(x) \leq \min_{z \in \mathcal{S}_j} V^*(z)$.

Proof of Claim 2. Let π^* be the optimal policy; accordingly $\pi^*(x)$ is fixed for every $x \in \mathcal{S}$. Then, for every $t > 0$, we can determine $x(t; x_1, \mathbf{u}_t^*, \mathbf{y}_t)$ for some sequence of disturbances \mathbf{y}_t , where the optimal input sequence $\mathbf{u}_t^* = \{u^*(0), \dots, u^*(t-1)\}$ (starting with x_1) can be recursively obtained as follows:

$$u^*(t) = \pi^*(x(t-1; x_1, \mathbf{u}_{t-1}^*, \mathbf{y}_{t-1})). \quad (\text{EC.11})$$

with the initialization $u^*(0) = \pi^*(x_1)$. For the above \mathbf{u}^* and \mathbf{y} , we can then determine the evolution of the states corresponding to initial state x_2 . Since $x(t; x_1, \mathbf{u}_t^*, \mathbf{y}_t) \geq x(t; x_2, \mathbf{u}_t^*, \mathbf{y}_t)$ by Claim 1, we notice readily that the reward $R_{u^*}(x(t; x_1, \mathbf{u}_t^*, \mathbf{y}_t)) \leq R_{u^*}(x(t; x_2, \mathbf{u}_t^*, \mathbf{y}_t))$ for every $t \geq 0$ (since the one-step reward is based only on the maximum delay, dwell time and control input, the inequality follows). Since the above holds for any given disturbance sequence, the expected discounted payoff associated with the state starting from x_1 i.e., $V^*(x_1)$, is no more than the expected discounted payoff associated with the state starting from x_2 , which we will denote by $V_{\mathbf{u}^*}(x_2)$. As a result, $V^*(x_1) \leq V_{\mathbf{u}^*}(x_2) \leq V^*(x_2)$. The second part of the inequality holds since \mathbf{u}_t^* as defined in (EC.11) is a sub-optimal control policy for the state evolution starting from x_2 and hence the expected discounted payoff associated with that policy is necessarily dominated by the optimal value function starting from x_2 . To complete the proof, consider two different partitions \mathcal{S}_i and \mathcal{S}_j such that $\mathcal{S}_i \geq \mathcal{S}_j$. Let $\bar{z} = \arg \min_{z \in \mathcal{S}_j} V^*(z)$ and this can always be found since we are dealing with a subset, \mathcal{S}_j of a finite state space \mathcal{S} . Since $\mathcal{S}_i \geq \mathcal{S}_j$, $\exists \bar{x} \in \mathcal{S}_i$ such that $\bar{x} \geq \bar{z}$. We have shown that for this case, $V^*(\bar{x}) \leq V^*(\bar{z}) = \min_{z \in \mathcal{S}_j} V^*(z) \Rightarrow \min_{x \in \mathcal{S}_i} V^*(x) \leq \min_{z \in \mathcal{S}_j} V^*(z)$. \square

Acknowledgments

This work was also partly supported by the AFRL Summer Faculty Program and AFOSR award no. FA9550-10-1-0392.

References

- Kish, B., M. Pachter, D. Jacques. 2009. *UAV Cooperative Decision and Control: Challenges and Practical Approaches*, chap. Effectiveness Measures for Operations in Uncertain Environments. *Advances in Design and Control*, SIAM, 103–124.
- Cover, Thomas M., Joy A. Thomas. 2006. *Elements of Information Theory*. 2nd ed. Wiley-Interscience.